Amendments to the Claims

Claim 1 (currently amended): A computer program product for improving scheduling of tasks, the computer program product embodied on one or more computer readable media and comprising:

computer-readable program code means for computing whether execution of a plurality of schedulable tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

computer-readable program code means for ~~adding~~ computing a task-specific maximum cost extension allowable for a subsequent execution of each task, wherein the maximum cost extensions for all the tasks, when taken together, allow ~~an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when~~ the execution of the plurality of tasks to remain ~~is computed to be~~ feasible; and

~~computer-readable program code means for iteratively repeating operation of the computer-readable program code means for computing and the computer-readable program code means for adding, until the execution is computed to be no longer feasible; and~~

computer-readable program code means for using, upon determining that any of the schedulable tasks exceeds its associated cost during its subsequent execution, the maximum cost extension ~~revised cost~~ for [[each]] that task as an upper limit on additional allowable execution time for the task~~, after operation of the computer-readable program code means for iteratively repeating~~.

Claim 2 (currently amended): The computer program product according to Claim 1, wherein the

maximum cost extension allowable for each task is computed as ~~additional amount of time is~~ a fixed percentage of the associated cost for the task.

Claim 3 (currently amended): The computer program product according to Claim 1, wherein the maximum cost extension allowable for each task is ~~additional amount of time is~~ zero for a subset of the tasks, and for all other tasks is computed as a fixed percentage of the associated cost for the task.

Claim 4 (currently amended): The computer program product according to Claim 1, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the computer-readable program code means for adding, the additional amount of time~~ is iteration, a fixed percentage of the associated cost for the task to compute a revised cost and by using, ~~wherein~~ on other iterations, ~~the additional amount of time is~~ a fixed percentage of the revised cost for the task.

Claim 5 (currently amended): The computer program product according to Claim 1, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the computer-readable program code means for adding, the additional amount of time~~ is iteration, zero for a subset of the tasks, and for all other tasks, [[is]] a fixed percentage of the associated cost for the task, thereby computing a revised cost for the task, and ~~wherein~~ by using, on other iterations, ~~the additional amount of time~~ is a fixed percentage of the revised cost for the task.

Claim 6 (currently amended): The computer program product according to Claim 1, wherein the computer-readable program code means for using further comprises:

computer-readable program code means for alternatively using ~~determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, computer-readable program code means for allowing the particular task to run until reaching a minimum of (1)~~ an amount of time remaining until the task's associated deadline as the upper limit on additional allowable execution time, when the time remaining is not greater than the maximum cost extension ~~or (2) the upper limit on execution time~~ for the task.

Claim 7 (currently amended): A system for improving scheduling of tasks, comprising:

at least one processor;

means for computing whether execution of a plurality of schedulable tasks by the at least one processor is feasible, wherein each of the tasks has an associated cost and an associated deadline;

means for computing a task-specific maximum cost extension allowable for a subsequent execution of each task, wherein the maximum cost extensions for all the tasks, when taken together, allow ~~adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when~~ the execution of the plurality of tasks to remain ~~is computed to be~~ feasible; and

means for using, upon determining that any of the schedulable tasks exceeds its associated cost during its subsequent execution, the maximum cost extension for that task as an

upper limit on additional allowable execution time for the task.

~~means for iteratively repeating operation of the means for computing and the means for adding, until the execution is computed to be no longer feasible.~~

Claim 8 (canceled)

Claim 9 (currently amended): The system according to Claim 7, wherein the maximum cost extension allowable for each task is computed as ~~additional amount of time is~~ a fixed percentage of the associated cost for the task.

Claim 10 (currently amended): The system according to Claim 7, wherein the maximum cost extension allowable for each task is ~~additional amount of time is~~ zero for a subset of the tasks, and for all other tasks is computed as a fixed percentage of the associated cost for the task.

Claim 11 (currently amended): The system according to Claim 7, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the means for adding, the additional amount of time is~~ iteration, a fixed percentage of the associated cost for the task to compute a revised cost and by using, ~~wherein~~ on other iterations, ~~the additional amount of time is~~ a fixed percentage of the revised cost for the task.

Claim 12 (currently amended): The system according to Claim 7, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the~~

means for adding, the additional amount of time is iteration, zero for a subset of the tasks, and for all other tasks, [[is]] a fixed percentage of the associated cost for the task, thereby computing a revised cost for the task, and wherein by using, on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

Claim 13 (currently amended): The system according to Claim 8, wherein the means for using further comprises:

means for alternatively using determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, means for allowing the particular task to run until reaching a minimum of (1) an amount of time remaining until the task's associated deadline as the upper limit on additional allowable execution time, when the time remaining is not greater than the maximum cost extension or (2) the upper limit on execution time for the task.

Claim 14 (currently amended): A computer-implemented method for improving scheduling of tasks, comprising steps of:

computing whether execution of a plurality of schedulable tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

computing a task-specific maximum cost extension allowable for a subsequent execution of each task, wherein the maximum cost extensions for all the tasks, when taken together, allow adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when the execution of the plurality of tasks to remain is computed to be feasible; and

upon determining that any of the schedulable tasks exceeds its associated cost during its subsequent execution, using the maximum cost extension for that task as an upper limit on additional allowable execution time for the task.

~~iteratively repeating operation of the computing step and the adding step, until the execution is computed to be no longer feasible.~~

Claim 15 (canceled)

Claim 16 (currently amended): The computer-implemented method according to Claim 14, wherein the maximum cost extension allowable for each task is computed as ~~additional amount of time is~~ a fixed percentage of the associated cost for the task.

Claim 17 (currently amended): The computer-implemented method according to Claim 14, wherein the maximum cost extension allowable for each task is ~~additional amount of time is~~ zero for a subset of the tasks, and for all other tasks is computed as a fixed percentage of the associated cost for the task.

Claim 18 (currently amended): The computer-implemented method according to Claim 14, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the adding step, the additional amount of time is~~ iteration, a fixed percentage of the associated cost for the task to compute a revised cost and by using, ~~wherein~~ on other iterations, ~~the additional amount of time is~~ a fixed percentage of the revised cost for the

task.

Claim 19 (currently amended): The computer-implemented method according to Claim 14, wherein the maximum cost extension allowable for each task is computed incrementally by using, on a first ~~iteration of the adding step, the additional amount of time is~~ iteration, zero for a subset of the tasks, and for all other tasks, [[is]] a fixed percentage of the associated cost for the task, thereby computing a revised cost for the task, and ~~wherein~~ by using, on other iterations, ~~the additional amount of time is~~ a fixed percentage of the revised cost for the task.

Claim 20 (currently amended): The computer-implemented method according to Claim 15, wherein the using step further comprises the steps of:

alternatively using ~~determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, allowing the particular task to run until reaching a minimum of (1)~~ an amount of time remaining until the task's associated deadline as the upper limit on additional allowable execution time, when the time remaining is not greater than the maximum cost extension ~~or (2) the upper limit on execution time~~ for the task.